

Compte rendu TP: Mes premiers pas en java

1.1 Installer la dernière version du JDK

J'ai installé la dernière version du JDK, qui est essentiel pour le développement Java. Il me permet de compiler et d'exécuter mes programmes Java.

Il contient :

- Le compilateur Java : qui transforme mon code source en bytecode.
- La JVM : qui exécute ce bytecode sur mon système d'exploitation, que ce soit Windows, Linux ou macOS.

1.2 Télécharger un JDK

Je me suis renseigné sur les différentes options pour télécharger et installer le JDK.

Traditionnellement, Java était disponible gratuitement via Oracle, mais depuis janvier 2019, la politique d'Oracle a changé et l'utilisation commerciale de la JDK Oracle n'est plus gratuite. Cela m'a poussé à chercher une alternative open-source.

Deux options principales s'offraient à moi :

- OpenJDK : le projet source de Java, qui contient toutes les dernières évolutions du langage. Mais OpenJDK ne fournit pas directement de versions exécutables ; il faut le compiler soi-même, ce qui n'est pas très pratique pour une installation rapide.

- AdoptOpenJDK (Adoptium) : un projet communautaire qui propose des versions exécutables du JDK pour Windows, Linux et macOS. C'est simple à installer et régulièrement mis à jour.

J'ai donc choisi AdoptOpenJDK (Adoptium), qui m'a permis d'installer facilement la JDK.

Étapes de l'installation :

- J'ai accédé au site [Adoptium](https://adoptopenjdk.net)
 - J'ai téléchargé et installé la version recommandée pour mon système d'exploitation.
-

1.3 Installer Eclipse

Je suis allé sur un navigateur (Google Chrome) sur le site [Eclipse.org](https://eclipse.org) et j'ai lancé le téléchargement puis l'installation en faisant attention à bien choisir la première option qui doit être "Java for Developer"

2. Créer un programme Java

L'objectif de cette étape était de créer un programme Java simple en utilisant un environnement de développement intégré (IDE). J'ai utilisé Eclipse, un des IDE les plus populaires pour le développement Java.

2.1 Ouvrir Eclipse

- J'ai lancé l'application Eclipse sur l'ordinateur.
- Lors de la première utilisation, Eclipse m'a demandé de spécifier un workspace (le répertoire où mes projets seraient sauvegardés). J'ai choisi un répertoire déjà existant ou j'en ai créé un nouveau selon mes besoins.
- L'interface d'Eclipse s'est affichée avec ses sections principales : la fenêtre d'édition du code, la vue des projets à gauche et la console en bas.

2.2 Créer un nouveau projet Java

- J'ai sélectionné File > New > Java Project.
 - J'ai donné un nom à mon projet, par exemple : *MonPremierProjetJava*.
 - J'ai laissé les options par défaut pour le JRE (puisque mon JDK était déjà configuré avec Eclipse).
 - J'ai cliqué sur Finish pour créer le projet.
 - Eclipse a généré la structure du projet avec un dossier src où je pouvais placer mon code.
-

2.3 Créer une classe Main

Ensuite, j'ai créé une classe principale appelée Main pour contenir la méthode main.

- J'ai cliqué droit sur le dossier src → New > Class.
- J'ai entré le nom Main.
- J'ai coché l'option public static void main(String[] args) pour que la méthode soit générée automatiquement.
- J'ai cliqué sur Finish, et Eclipse a créé la classe Main avec la méthode main prête à l'emploi.

La méthode main est le point d'entrée du programme. Le paramètre String[] args permet de passer des arguments en ligne de commande, même si je ne l'ai pas encore utilisé.

J'ai ensuite vérifié que le fichier Main.java était bien dans le dossier src et qu'aucune erreur n'apparaissait dans la fenêtre *Problems*.

3.2 Saisir une variable

J'ai ajouté une variable dans ma classe Main pour tester mon code Java et comprendre comment gérer des données simples.

```
1 package projet;
2
3 import java.util.Scanner;
4
5 public class maclasse {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.println("Veuillez saisir un mot :");
9         String str = sc.nextLine();
10        System.out.println("Vous avez saisi : " + str);
11        sc.close();
12    }
13 }
14
```

4. Lancer le programme

Enfin, j'ai lancé le programme directement depuis Eclipse(Run). Le code a été compilé et exécuté automatiquement, et le résultat s'est affiché dans la console.

